Formal Approaches for Swarm Technology (FAST)

Code 588 / Walt Truszkowski

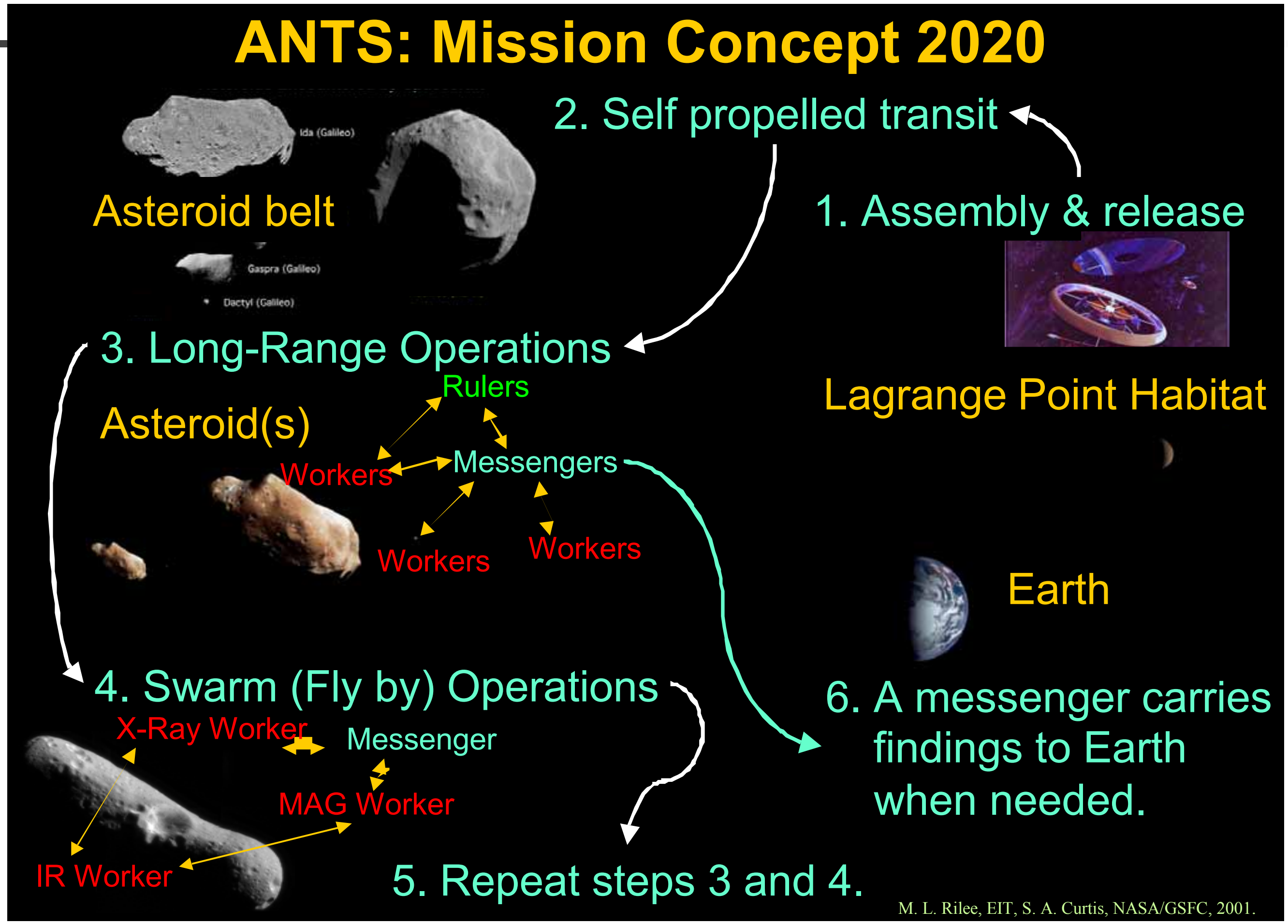
The Information Systems Division

Formal Approaches to Swarm Technology (FAST)

- The objective is to formally model a swarm system to be able to test for completeness and correctness of system behaviors

- The Autonomous Nano Technology Swarm (ANTS) concept was selected as domain of investigation

Focus of Formal Modeling



Difficulties in Formal Modeling of Swarms

- Emergent properties that may not be known
- Highly distributed and parallel
- Large number of interacting entities
- Worse than exponential growth in interactions
- Intelligent entities (capabilities increase over time)
- Total or near total autonomy
- Very little experience in verification and validation of swarm-based systems

Formal Models Considered

- **CSP –** Communicating Sequential Processes
- **X-Machines**
- **WSCCS –** Weighted Synchronous Calculus of Communicating Systems
- **Unity logic**

FAST - CSP

- Communicating Sequential Processes
  - Developed by C.A.R. Hoare
  - Process Algebra
  - Processes are recursively defined as an event followed by a process
  - Use channels for processes to communicate
  - Channels are guarded by events
  - Properties of CSP specifications can be proven correct

FAST - CSP

- **Evaluation**
  - Strengths
    - Specifying inter-process protocols
    - Identifying race conditions
    - Easily translated into model checking language
  - Weaknesses
    - No mechanism for analyzing emergent behavior

FAST - X-Machines

- Developed by Samuel Eilenberg
- Based on Finite State Machines
- Have an internal memory state
- Transitions are functions:
  - Input x Memory $\rightarrow$ Output x Memory

FAST – X-Machines

- **Evaluation**
  - Strengths
    - State-based system with memory
    - Executable
  - Weaknesses
    - No robust means for reasoning about or predicting behaviors beyond standard propositional logic

FAST - WSCCS

- **Weighted Synchronous Calculus of Communicating Systems**
  - Developed by Chris Tofts
  - Used to model social insects
  - Uses a state-based approach with frequencies and priorities for transitions
  - Frequencies and priorities can give you the probability of taking an action

FAST - WSCCS

- **Evaluation**
  - Strengths
    - Process algebra strengths
    - Specifies priorities and probabilities of actions
    - Rules for predicting behavior
  - Weaknesses
    - Inability to track and model goals and other aspects of the spacecraft

FAST - Unity Logic

- Developed by Chandy and Misra
- Syntax similar to Propositional Logic
- Reason about predicates and states
- Very good for concurrent systems

FAST – Unity Logic

- **Evaluation**
  - Strengths
    - Strengths of logic-based systems
    - A method for reasoning about predicates and the states they imply
    - A method for defining specific mathematical and statistical calculations to be performed
  - Weaknesses
    - Not rich enough for reasoning about emergent behavior

Formal Models – Next Steps

- Blending of methods seems to be the best approach for specifying swarm-based systems
- Blending of priority and probability aspects of WSCCS with memory and transition function aspects of X-Machines as well as logic from Unity Logic is a possibility and is currently being studied